

**Amendments to the Claims:**

This listing of claims will replace all prior versions and listings of claims in the application:

**Listing of Claims:**

Claim 1 (currently amended): A method in a computer system for isolating and teaching concepts of a programming language, comprising:

providing an interpreter interface including a code entry portion adapted for receiving input from a user;

receiving a single code entry comprising a programming statement in the programming language from the user via the code entry portion;

processing the code entry, wherein the processing includes comparing the code entry to a set of syntax and language rules for the programming language to identify errors by comparing syntax of the code entry to a set of syntax rules for the programming language to identify a syntax error or validity of the code entry syntax and when no errors are identified, executing the code entry, and when the comparing identifies the syntax error, retrieving an error code based on the syntax error; and

in response to the processing, displaying a visual cue to the user wherein the visual cue includes the error code when comparing identifies the syntax error and a semantic view of effects including created variables of executing the code entry when no syntax errors are identified.

Claim 2 (cancelled):

Claim 3 (cancelled):

Claim 4 (currently amended): The method of Claim [[3]]1, wherein the visual cue further includes the received code entry.

Claim 5 (original): The method of Claim 4, wherein the interpreter interface includes a code entry history portion for displaying the error code, the received code entry, and previously received and processed code entries.

Claim 6 (cancelled)

Claim 7 (cancelled):

Claim 8 (currently amended): The method of Claim [[7]]1, wherein the semantic view includes displaying a type, a name, and a value of a variable declared and assigned in the code entry.

Claim 9 (original): The method of Claim 1, wherein the displaying includes displaying objects created by or manipulated by execution of the code entry.

Claim 10 (original): The method of Claim 1, wherein the displaying includes displaying arrays created by or manipulated by execution of the code entry.

Claim 11 (currently amended): A computer program product for isolating and teaching concepts of a programming language, comprising:

first computer code devices configured to cause a computer to receive a programming statement from a user;

second computer code devices configured to cause a computer to process the programming statement including comparing the programming statement to syntax rules for the programming language to identify presence or absence of a syntax error in the programming statement;

third computer code devices configured to cause a computer to display a visual cue to the user based on the identified presence or absence of the syntax error; [[and]]

fourth computer code devices configured to cause a computer to retrieve an error code based on the identified syntax error and wherein the visual cue includes the retrieved error code;

fifth computer code devices for causing a computer to execute the programming statement when the second computer code devices identifies the absence of the syntax error; and

sixth computer code devices for displaying to the user a semantic view of effects of executing by the fifth computer code devices including created variable values.

Claim 12 (cancelled)

Claim 13 (original) The computer program product of Claim 11, wherein the first computer code devices are further configured to provide an interpreter interface including a code entry portion for performing the receiving of the programming statement and a code entry history portion for displaying at least part of the visual cue, the received programming statement, and previously received and processed programming statements.

Claim 14 (cancelled):

Claim 15 (cancelled):

Claim 16 (currently amended): A computer system for teaching programming language concepts, comprising:

means for receiving a code entry comprising a single programming statement in a programming language from a user;

a syntax validator processing the received code entry based on syntax rules for the programming language to determine syntax validity or a syntax error; and  
a semantic view engine displaying a semantic view to the user based on the determined syntax validity or the syntax error, wherein the semantic view includes effects of execution of the code entry including created variable values, ~~object configurations,~~ and arrays.

Claim 17 (original): The computer system of Claim 16, wherein the semantic view includes an error statement selected based on the syntax error.

Claim 18 (cancelled)

Claim 19 (original): The computer system of Claim 16, further including an execution engine operating to execute the code entry when the syntax is determined valid, wherein the code entry is executed based on a previously entered code entry.

Claim 20 (original): The computer system of Claim 16, wherein the receiving means includes an interface for displaying an interpreter interface with a code entry window for accepting the code entry by the user and for displaying a code entry history having previously-entered code entries and error statements corresponding to the previously-entered code entries.